What is claimed is:

1. A method comprising:

multiplying a dividend value by a first value to generate a second value associated with a product, wherein the first value is associated with a scaled approximate reciprocal of a divisor value, and wherein the scaled approximate reciprocal of the divisor value includes a compound exponent value;

extracting a third value from the second value, wherein the third value is generated using at least a subset bitfield of the second value; and

determining a remainder value based on the third value.

2. A method as defined in claim 1, wherein the dividend value and the divisor value are non-negative integers.

3. A method as defined in claim 1, wherein determining the remainder value includes calculating a scaling value by raising two to the power of the compound exponent value.

4. A method as defined in claim 3, wherein determining the remainder value includes calculating a fourth value by adding the scaling value to a compensation value associated with a rounding-up process.

5. A method as defined in claim 4, wherein determining the remainder value includes calculating the first value by dividing the fourth value by the divisor value.

6. A method as defined in claim 1, wherein the second value is stored in a range of contiguous bits.

7. A method as defined in claim 6, wherein the range of contiguous bits includes a residuary subset of contiguous bits, and wherein the residuary subset of contiguous bits includes the third value.

8. A method as defined in claim 1, wherein determining the remainder value includes setting the remainder value equal to the third value.

9. A method as defined in claim 1, wherein determining the remainder value includes setting the remainder value equal to zero if the third value equals zero.

10. A method as defined in claim 1, wherein determining the remainder value includes determining that the dividend value is exactly divisible by the divisor value if the third value equals zero.

11. A method as defined in claim 1, wherein determining the remainder value includes locating the remainder value stored within a data structure location associated with the third value.

12. A method as defined in claim 1, wherein determining the remainder value includes calculating the remainder value by adding, multiplying and bit-shifting, wherein adding results in a sum associated with the third value and one, wherein multiplying produces a product of the divisor value and the sum, and wherein bit-shifting includes right bit-shifting the product of the divisor value and the sum.

13. An apparatus comprising:

a processor system including a memory;

instructions stored in the memory that enable the processor system to:

multiply a dividend value by a first value to generate a second value associated with a product, wherein the first value is associated with a scaled approximate reciprocal of a divisor value, and wherein the scaled approximate reciprocal of the divisor value includes a compound exponent value;

extract a third value from the second value, wherein the third value is generated using at least a subset bitfield of the second value; and

determine a remainder value based on the third value.

14. An apparatus as defined in claim 13, wherein the dividend value and the divisor value are non-negative integers.

15. An apparatus as defined in claim 13, wherein the instructions enable the processor system to calculate a scaling value by raising two to the power of the compound exponent value.

16. An apparatus as defined in claim 15, wherein the instructions enable the processor system to calculate a fourth value by adding the scaling value to a compensation value associated with a round-up process.

17. An apparatus as defined in claim 16, wherein the instructions enable the processor system to calculate the first value by dividing the fourth value by the divisor value.

18. An apparatus as defined in claim 13, wherein the second value is stored in a range of contiguous bits.

19. An apparatus as defined in claim 18, wherein the range of contiguous bits includes a residuary subset of contiguous bits, and wherein the third value is stored in the residuary subset of contiguous bits.

20. An apparatus as defined in claim 13, wherein the remainder value is equal to the third value.

21. An apparatus as defined in claim 13, wherein the remainder value is equal to zero if the third value equals zero.

22. An apparatus as defined in claim 13, wherein the instructions enable the processor system to determine that the dividend value is exactly divisible by the divisor value if the third value equals zero.

23. An apparatus as defined in claim 13, wherein the instructions enable the processor system to locate the remainder value stored within a data structure location associated with the third value.

24. An apparatus as defined in claim 13, wherein the instructions enable the processor system to calculate the remainder value using an addition operation, a multiplication operation and a bit shift operation, wherein the addition operation results in a sum associated with the third value and one, wherein the multiplication operation produces a product of the divisor value and the sum, and wherein the bit-shift operation includes right bit-shifting the product of the divisor value and the sum.

25. A computer readable medium having instructions stored thereon that, when executed, cause a machine to:

multiply a dividend value by a first value to generate a second value associated with a product, wherein the first value is associated with a scaled approximate reciprocal of a divisor value, and wherein the scaled approximate reciprocal of the divisor value includes a compound exponent value;

extract a third value from the second value, wherein the third value is generated using at least a subset bitfield of the second value; and

determine the remainder value based on the third value.

26. A computer readable medium as defined in claim 25, wherein the dividend value and the divisor value are non-negative integers.

27. A computer readable medium as defined in claim 25 having instructions stored thereon that, when executed, cause the machine to calculate a scaling value by raising two to the power of the compound exponent value.

28. A computer readable medium as defined in claim 27 having instructions stored thereon that, when executed, cause the machine to calculate a fourth value by adding the scaling value to a compensation value associated with a round-up process.

29. A computer readable medium as defined in claim 27 having instructions stored thereon that, when executed, cause the machine to calculate the first value by dividing the fourth value by the divisor value.

30. A computer readable medium as defined in claim 25, wherein the second value is stored in a range of contiguous bits.

31. A computer readable medium as defined in claim 30, wherein the range of contiguous bits includes a residuary subset of contiguous bits, and wherein the third value is stored in the residuary subset of contiguous bits.

32. A computer readable medium as defined in claim 25, wherein the remainder value is equal to the third value.

33. A computer readable medium as defined in claim 25, wherein the remainder value is equal to zero if the third value equals zero.

34. A computer readable medium as defined in claim 25 having instructions stored thereon that, when executed, cause the machine to determine that the dividend value is exactly divisible by the divisor value if the third value equals zero.

35. A computer readable medium as defined in claim 25 having instructions stored thereon that, when executed, cause a machine to locate the remainder value stored within a data structure location associated with the third value.

36. A computer readable medium as defined in claim 25 having instructions stored thereon that, when executed, cause the machine to calculate the remainder value using an addition operation, a multiplication operation and a bit shift operation, wherein the addition operation results in a sum associated with the third value and one, wherein the multiplication operation produces a product of the divisor value and the sum, and wherein the bit-shift operation includes right bit-shifting the product of the divisor value and the sum.

37. A method comprising:

calculating a first value associated with a scaled approximate reciprocal of a divisor value;

calculating a second value, wherein the second value is a product of a dividend value and the first value, and wherein the second value is stored in a first range of contiguous bits;

extracting a third value from a second range of contiguous bits that lies within the first range of contiguous bits; and

computing a remainder value based on the third value.

38. A method as defined in claim 37, wherein the first value is associated with a compound exponent value and a compensation value.

39. A method as defined in claim 38, wherein the compound exponent value is associated with upper and lower bit positions of the second range of contiguous bits.

40. A method as defined in claim 38, wherein the compensation value is associated with a round-up process.

41. A method as defined in claim 37, wherein computing the remainder value includes calculating the remainder value by adding, multiplying and bit-shifting.

42. A method as defined in claim 37, wherein computing the remainder value includes locating the remainder value in a data structure, and wherein the third value is associated with a location reference to an entry in the data structure.

43. A system comprising:

a processor system including a memory; and

instructions stored in the memory that enable the processor system to:

calculate a first value associated with a scaled approximate reciprocal of a divisor value;

calculate a second value, wherein the second value is a product of a dividend value and the first value, and wherein the second value is stored in a first range of contiguous bits;

extract a third value from a second range of contiguous bits that lies within the first range of contiguous; and

compute a remainder value based on the third value.

44. A system as defined in claim 43, wherein the first value is associated with a compound exponent value and a compensation value.

45. A system as defined in claim 44, wherein the compound exponent value is associated with upper and lower bit positions of the second range of contiguous bits.

46. A system as defined in claim 44, wherein the compensation value is associated with a round-up process.

47. A system as defined in claim 43, wherein the instructions enable the processor system to calculate the remainder value using at least one of an addition operation, a multiplication operation and a bit-shift operation.

48. A system as defined in claim 43, wherein the instructions enable the processor system to locate the remainder value in a data structure, wherein the third value is associated with a location reference to an entry in the data structure.

49. A computer readable medium having instructions stored thereon that, when executed, cause a machine to:

calculate a first value associated with a scaled approximate reciprocal of a divisor value;

calculate a second value, wherein the second value is a product of a dividend value and the first value, and wherein the second value is stored in a first range of contiguous bits;

extract a third value from a second range of contiguous bits that lies within the first range of contiguous bits includes; and

compute a remainder value based on the third value.

50. A computer readable medium as defined in claim 49, wherein the first value is associated with a compound exponent value and a compensation value.

51. A computer readable medium as defined in claim 50, wherein the compound exponent value is associated with upper and lower bit positions of the second range of contiguous bits.

52. A computer readable medium as defined in claim 50, wherein the compensation value is associated with a round-up process.

53. A computer readable medium as defined in claim 49 having instructions stored thereon that, when executed, cause the machine to calculate the remainder value using at least one of an addition operation, a multiplication operation and a bit-shift operation.

54. A computer readable medium as defined in claim 49 having instructions stored thereon that, when executed, cause the machine to locate the remainder value in a data structure, wherein the third value is associated with a location reference to an entry in the data structure.

55. A method for computing a remainder value comprising:

extracting a first range of contiguous bits from a second range of contiguous bits, wherein the first range of contiguous bits is bound by an upper-limit bit position value and a lower-limit bit position value, and wherein the first range of contiguous bits is associated with an intermediate remainder calculating value; and

computing the remainder value based on the intermediate remainder calculating value.

56. A method as defined in claim 55, wherein computing the remainder value includes calculating the remainder value by adding, multiplying and bit-shifting, wherein adding results in a sum associated with the intermediate remainder calculating value and one, wherein multiplying produces a product of a dividend value and the sum, and wherein bit-shifting consists of right bit-shifting the product of a dividend value and the sum.

57. A method as defined in claim 55, wherein computing the remainder value includes setting the remainder value equal to zero if the intermediate remainder calculating value equals zero.

58. A method as defined in claim 55, wherein computing the remainder value includes determining that a dividend value is exactly divisible by a divisor value if the intermediate remainder calculating value equals zero.

59. A method as defined in claim 55, wherein computing the remainder value includes locating the remainder value in a data structure, wherein the intermediate remainder calculating value is associated with a location reference to an entry in the data structure.

60. A method as defined in claim 55, wherein computing the remainder value includes determining that the remainder value is equal to the intermediate remainder calculating value.

61. A system comprising:

a processor system including a memory; and

instructions stored in the memory that enable the processor system to:

extract a first range of contiguous bits from a second range of contiguous bits, wherein the first range of contiguous bits is bound by an upper-limit bit position value and a lower-limit bit position value, and wherein the first range of contiguous bits is associated with an intermediate remainder calculating value; and

compute a remainder value based on the intermediate remainder calculating value.

62. A system as defined in claim 61, wherein the instructions enable the processor system to calculate the remainder value using an addition operation, a short multiplication operation and a bit-shift operation, wherein the addition operation results in a sum associated with the intermediate remainder calculating value and one, wherein the short multiplication operation produces a product of a dividend value and the sum, and wherein the bit-shift operation consists of right bit-shifting the product of a dividend value and the sum.

63. A system as defined in claim 61, wherein the instructions enable the processor system to determine that the remainder value is equal to zero if the intermediate remainder calculating value equals zero.

64. A system as defined in claim 61, wherein the instructions enable the processor system to determine that a dividend value is exactly divisible by a divisor value if the intermediate remainder calculating value equals zero.

65. A system as defined in claim 61, wherein the instructions enable the processor system to locate the remainder value in a data structure, wherein the intermediate remainder calculating value is associated with a location reference to an entry in the data structure.

66. A system as defined in claim 61, wherein the instructions enable the processor system to determine that the remainder value is equal to the intermediate remainder calculating value.

67. A computer readable medium having instructions stored thereon that, when executed, cause a machine to:

extract a first range of contiguous bits from a second range of contiguous bits, wherein the first range of contiguous bits is bound by an upper-limit bit position value and a lower-limit bit position value, and wherein the first range of contiguous bits is associated with an intermediate remainder calculating value; and

compute the remainder value based on the intermediate remainder calculating value.

68. A computer readable medium as defined in claim 67, having instructions stored thereon that, when executed, cause the machine to calculate the remainder value using an addition operation, a short multiplication operation and a bit-shift operation, wherein the addition operation results in a sum associated with the intermediate remainder calculating value and one, wherein the short multiplication operation produces a product of a dividend value and the sum, and wherein the bit-shift operation consists of right bit-shifting the product of a dividend value and the sum.

69. A computer readable medium as defined in claim 67, having instructions stored thereon that, when executed, cause the machine to determine that the remainder value is equal to zero if the intermediate remainder calculating value equals zero.

70. A computer readable medium as defined in claim 67, having instructions stored thereon that, when executed, cause the machine to determine that a dividend value is exactly divisible by a divisor value if the intermediate remainder calculating value equals zero.

71. A computer readable medium as defined in claim 67, having instructions stored thereon that, when executed, cause the machine to locate the remainder value in a data structure, wherein the intermediate remainder calculating value is associated with a location reference to an entry in the data structure.

72. A computer readable medium as defined in claim 67, having instructions stored thereon that, when executed, cause the machine to determine that the remainder value is equal to the intermediate remainder calculating value.

73. A system comprising:

a bit extractor; and

a remainder value generator coupled to the bit extractor, wherein the bit extractor is configured to extract a residuary subset bitfield associated with an intermediate remainder calculating value and a compound exponent value, and wherein the remainder value generator is configured to generate a remainder value that is associated with the intermediate remainder calculating value.

74. A system as defined in claim 73, wherein the residuary subset bitfield is associated with an upper-boundary bit position value and a lower-boundary bit position value, and wherein the compound exponent value includes the upper-boundary bit position value and the lower-boundary bit position value.

75. A system as defined in claim 73, wherein the remainder value generator is configured to generate the remainder value using at least one of an addition operation, a multiplication operation and a bit-shift operation.

76. A system as defined in claim 73, wherein the remainder value generator is configured to generate a remainder value equal to zero if the intermediate remainder calculating value equals zero.

77. A system as defined in claim 73, wherein the remainder value generator is configured to locate the remainder value stored within a data structure location associated with the intermediate remainder calculating value.